

A Bayesian Approach for 3D Models Retrieval Based on Characteristic Views

Tarik Filali Ansary, Jean-Philippe Vandeborre, Mohamed Daoudi
Equipe MIIRE (INT / LIFL)

ENIC Telecom Lille 1
Cité Scientifique – rue Guglielmo Marconi
59658 Villeneuve d'Ascq cedex – France
{filali,vandeborre,daoudi}@enic.fr

Abstract

The management of big databases of three-dimensional models (used in CAD applications, visualization, games, etc.) is a very important domain. The ability to characterize and easily retrieve models are a key issues for the designers and the final users. In this frame, two main approaches exist: search by example of a three-dimensional model, and search by a 2D view. In this paper we focus on the characterization of a 3D model by a set of views (called characteristic views), and on the indexing process of these models with a Bayesian probabilistic approach using the characteristic views. We illustrate our results using a collection of three-dimensional models supplied by Renault Group.

1. Introduction

The use of three-dimensional images and models databases throughout the Internet is growing both in number and in size. The development of modeling tools, 3D scanners, 3D graphic accelerated hardware, Web3D and so on, is enabling access to three-dimensional materials of high quality. In recent years, many systems have been proposed for efficient information retrieval from digital collections of images and videos. However, the solutions proposed so far to support retrieval of such data are not always effective in application contexts where the information is intrinsically three-dimensional.

A similarity metric has to be defined to compute a visual similarity between two 3D models, given their descriptions. Two families of methods for 3D model retrieval exist: 3D/3D (direct model analysis) and 2D/3D (3D model analysis from its 2D views) retrieval.

For example, Vandeborre et al. [1] propose to use full three-dimensional information. The 3D objects are represented as mesh surfaces and 3D shape descriptors are used. The results obtained show the limitation of the approach when the mesh is not regular. This kind of approach is not robust in terms of shape representation.

In 2D/3D retrieval approach, two serious problems arise: how to characterize a 3D model with few 2D views, and how to use these views to retrieve the model from a 3D models collection.

Abbasi and Mokhtarian [2] propose a method that eliminates the similar views in the sense of a distance among CSS (*Curvature Scale Space*) from the outlines of these views. At last, the minimal number of views is selected with an optimization algorithm. Dorai and Jain [3] use, for each of the model in the collection, an algorithm to generate 320 views. Then, a hierarchical classification, based on a distance measure between curvature histogram from the views, follows.

Mahmoudi and Daoudi [4] also suggest to use the CSS from the outlines of the 3D model extracted views. The CSS is then organized in a tree structure called *M-tree*. Chen & Stockman [5] as well as Yi and al. [6] propose a method based on a bayesian probabilistic approach. It means computing an *a posteriori* probability to recognize the model when a certain feature is observed. This probabilistic method gives good results, but the method was tested on a small collection of 20 models.

In this paper, we propose a method for optimal selection of 2D views from a 3D model, and a probabilistic method for 3D models indexing from these views. This paper is organized in the following way. In section 2, we present the principles of our algorithm for characteristic views selection. In section 3, a probabilistic approach for 3D retrieval is proposed. Finally, the results obtained from a collection of 3D models are presented.

2. Selection of characteristic views

In this paragraph, we present our algorithm for optimal characteristic views selection from a three-dimensional model.

2.1. Computing the set of views

Let $D_b = \{M_1, M_2, \dots, M_N\}$ be a collection of N three-dimensional models. We wish to represent a 3D model M_i of this collection by a set of 2D views.

For each model M_i of the collection, 2D views are generated from multiple viewpoints. These viewpoints are equally spaced on the unit sphere. In our current implementation, there are 80 views. We subdivided the icosahedron once by using the *Loop's* subdivision scheme to generate the 80 faceted polyhedron. Then 80 cameras

are placed at each face-center looking at the coordinate origin.

Discrimination of characteristic views

Let $V_M = \{V_M^1, V_M^2, \dots, V_M^v\}$ be the set of 2D views from the three-dimensional model M , where v is the total number of views.

Among this set of views, we have to select those that characterize effectively the three-dimensional model according to a feature of these views. The chosen feature, even though our method is **independent** from this one, is the curvatures histogram.

Curvatures histogram of a view

We use the histogram of three-dimensional shape introduced by Koenderink and Van Doorn [7] to describe a 2D view. First of all, the 2D view is transformed into a gray-level image to obtain a three-dimensional surface in which every pixel is a point of the surface. The Z coordinate of the points is then equal to the gray-intensity of the considered pixel $(x, y, I(x, y))$.

This index of three-dimensional shape was particularly used for the indexing process of fixed images [8], depth images [3], and three-dimensional models [1][9].

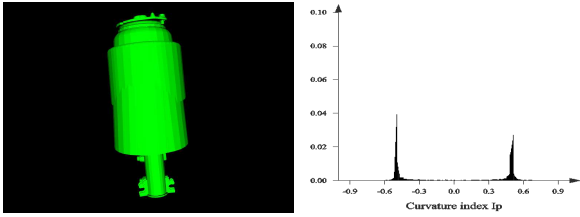


Figure1. Curvature histogram of a 2D view.

Comparison of views

Each view is then described with its curvatures histogram. To compare two views, it is enough to compare their respective histograms. There are several ways to compare distribution histograms: the Minkowski Ln norms, Kolmogorov-Smirnov distance, Match distances, and many others. We choose to use the L1 norm because of its simplicity and its accurate results.

$$D_{L_1}(f_1, f_2) = \left(\int_{-\infty}^{+\infty} |f_1 - f_2| \right)$$

The calculation of the distance between the histograms of two views can be then considered as the distance between these views.

Selection of characteristic views

The next step is to reduce the set of views of a model M to a set that represents only the most important views. This set is called the set of characteristic views V_{C_M} .

A view V_M^k is a characteristic view of a model M for a distance ϵ , if the distance between this view and all the

other characteristic views of M is greater than ϵ . That is to say:

$$V_M^j \in V_{C_M} \Leftrightarrow \forall V_{C_M}^k \in V_{C_M}, D_{V_M^j, V_{C_M}^k} > \epsilon$$

With $D_{V_M^j, V_{C_M}^k}$ the distance between the curvatures

histograms of V_M^j and $V_{C_M}^k$.

However, the choice of the distance threshold ϵ is important and depends on the complexity of the three-dimensional model. This information is not *a priori* known.

To solve the problem of determining the distance threshold ϵ , we adapte the previous algorithm by taking into account an interval of these distances from 0 to 1 with a step of 0.001. The final set of characteristic views is then the union of all the sets of characteristic views for every ϵ in $[0..1]$.

2.2. Properties of the views selection algorithm

To reduce the number of characteristic views, we filter this set of views in a way that for each model it verifies two criterions:

- Each view of the model M must be represented by at least one characteristic view. This means:

$$\forall V_M^j \in V_M, \exists V_{C_M}^k \text{ such as } \mathfrak{R}(V_M^j) = V_{C_M}^k$$

With \mathfrak{R} an **application** associating to each element of V_M an elements of V_{C_M} ;

- Characteristic views must not be redundant.

Let $V_{R_M}^j$ be the set of views represented by the characteristic view $V_{C_M}^j$. A characteristic view $V_{C_M}^j$ is said redundant if there is a set of characteristic views for which the union of represented views includes $V_{R_M}^j$.

$$V_{R_M}^j \subset \bigcup V_{R_M}^k$$

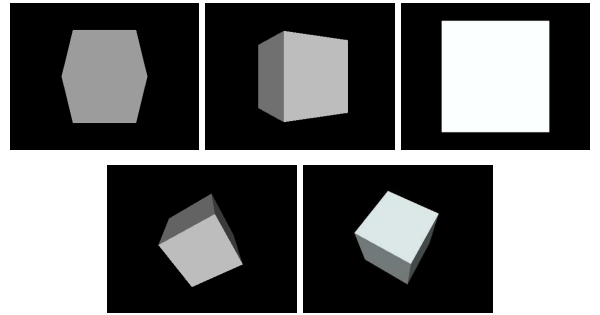


Figure 2. Characteristic views of a cube.

3. Probabilistic approach for 3D indexing

Each model of the collection D_b is represented by a set of characteristic views $V_C = \{V_{C_1}, V_{C_2}, \dots, V_{C_{\hat{v}}}\}$, with \hat{v} the number of characteristic views. To each

characteristic view corresponds a set of represented views called Vr .

Considering a request 2D view Q , we wish to find the model $M_i \in Db$ which one of its characteristic views is the closest to the request view Q . This model is the one that has the highest probability $P(M_i, Vc_{M_i}^j / Q)$.

Let H be the set of all the possible hypotheses of correspondence between the request view Q and a model M , $H = \{h_1 \vee h_2 \vee \dots \vee h_{\hat{v}}\}$. A hypothesis h_k means that the view k of the model is the view request Q . The sign \vee represents *logic or operator*. Let us note that if an hypothesis h_k is true, all the other hypotheses are false.

$P(M_i, Vc_{M_i}^j / Q)$ can be expressed by $P(M_i, Vc_{M_i}^j / H)$.

The closest model is the one that contains a view having the highest probability. Using the Bayes theorem, we have:

$$P(M_i, Vc_{M_i}^j / H) = \frac{P(H, Vc_{M_i}^j / M_i)P(M_i)}{P(H)}$$

We also have:

$$P(H, Vc_{M_i}^j / M_i) = \sum_{k=1}^{\hat{v}} P(h_k, Vc_{M_i}^j / M_i)$$

The sum $\sum_{k=1}^{\hat{v}} P(h_k, Vc_{M_i}^j / M_i)$ can be reduced to the only true hypothesis $P(h_j, Vc_{M_i}^j / M_i)$.

By integrating this remark, we obtain:

$$P(M_i, Vc_{M_i}^j / H) = \frac{P(h_j / Vc_{M_i}^j, M_i)P(Vc_{M_i}^j / M_i)P(M_i)}{\sum_{i=1}^N \sum_{j=1}^{\hat{v}} P(h_j / Vc_{M_i}^j, M_i)P(Vc_{M_i}^j / M_i)P(M_i)}$$

With $P(M)$ the probability to observe the model M .

$$P(M_i) = \alpha e^{\left(-\alpha \frac{N(Vc_{M_i})}{N(Vc)}\right)}$$

Where $N(Vc_{M_i}^j)$ is the number of characteristic views of the model M , and $N(Vc)$ is the total number of characteristic views for the set of the models of the collection Db . α is a parameter to hold the effect of the probability $P(M_i)$. The algorithm conception makes that, the greater the number of characteristic views of an object, the more it is complex. Indeed, simple object (e.g. a cube) can be at the root of more complex objects. On the other hand:

$$P(Vc_{M_i}^j / M_i) = 1 - \beta e^{\left(-\beta \frac{N(Vr_{M_i}^j)}{N(Vr_{M_i})}\right)}$$

Where $N(Vr_{M_i}^j)$ is the number of views represented by

the characteristic view j of the model M , and $N(Vr_{M_i})$ is the total number of views represented by the model M_i . Coefficient β is introduced to reduce the effect of the view probability. We use the values $\alpha = \beta = 1/100$ which give the best results during our experiments. The greater is the number of represented views $N(Vr_{M_i}^j)$, the more the characteristic view $Vc_{M_i}^j$ is important and the best it represents the three-dimensional model.

The value $P(h_j / Vc_{M_i}^j, M_i)$ is the probability that, knowing that we observe the characteristic view j of the model M_i , this view is the request view Q :

$$P(h_j / Vc_{M_i}^j, M_i) = e^{(-D_{h_q, h_{Vc_{M_i}^j}})}$$

With $D_{h_q, h_{Vc_{M_i}^j}}$ the Minkowski distance between the

curvatures histograms of Q and of the $Vc_{M_i}^j$ characteristic view of the three-dimensional model M_i .

4. Experiments and results

We implemented our algorithm, described in the previous sections, using C++ and the TGS OpenInventor.

To measure the performance, we classified the 132 models of our collection into 14 classes based on the judgment of two adult persons.

We used several different performance measures to objectively evaluate our method: The First Tier (FT), the second Tier (ST), and Nearest Neighbor (NN) match percentages, as well as the recall-precision plot.

Recall and precision are well known in the literature of content-based search and retrieval. The recall and precision are defined as follow:

$$\text{Recall} = N/Q, \text{ Precision} = N/A$$

With N the number of relevant models retrieved in the top A retrievals. Q is the number of relevant models in the collection, that is, the class number of models to which the query belongs to.

FT, ST, and NN percentages are defined as follows. Assume that the query belongs to the class C containing Q models. The FT percentage is the percentage of the models from the class C that appeared in the top $(Q-1)$ matches.

The ST percentage is similar to FT, except that it is the percentage of the models from the class C the top $2(Q-1)$ matches. The NN percentage is the percentage of the cases in which the top matches are drawn from the class C .

We test the performance of our method with and without the use of the probabilistic approach. Table 1 shows performance in terms of the FT, ST, and NN. Figure 3 shows the recall precision plots, we can notice the contribution of the probabilistic approach to the improvement of the result. To produce results, we queried a random model from each class. Five random views were taken from every model selected. Results are the average of 70 queries

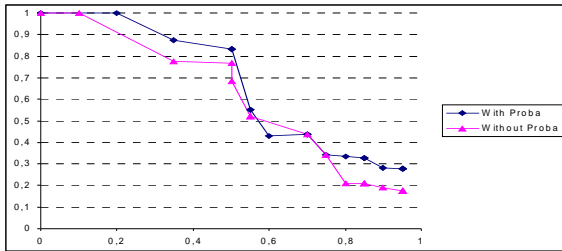


Figure 3. Overall recall precision.

Methods	Performance		
	FT	ST	NN
With Proba	28.88	40.65	51.46
Without Proba	27.84	37.26	47.18

Table1. Retrieval performance.

Figure 4 and 5 show an example of querying by using the methods with probabilistic approach.



Figure 4. Request input is a random view of a model.

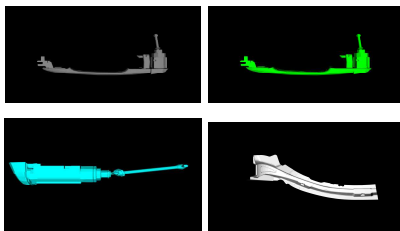


Figure 5. Top 4 retrieved models from the collection.

5. Conclusion and future work

We have presented a new method to extract characteristic views from a 3D model. This method is independent from the feature used to describe the view. And we also proposed a bayesian probabilistic method for 3D models retrieval from a single random view.

Our algorithm for characteristic views extraction let us characterize a 3D model by a small number of views.

Our method is robust in terms of the shape representations it accepts. The method can be used against topologically ill defined mesh-based models, e.g., polygon

soup models. This is because the method is appearance based: practically any 3D model can be in the database.

The evaluation experiments showed that the method gives very satisfactory results. In the retrieval experiments, it performs significantly better when we use the probabilistic approach for retrieval.

6. Acknowledgments

This work is supported by the French Research Ministry and the RNRT (Réseau National de Recherche en Télécommunications) within the framework of the Semantic-3D national project (<http://www.semantic-3d.net>).

7. Bibliography

- [1] J.P. Vandeborre, V. Couillet and M. Daoudi, "A Practical Approach for 3D Model Indexing by combining Local and Global Invariants", 3D Data Processing Visualization and Transmission (3DPVT), pp. 644-647, Padova, Italy, June 2002.
- [2] S. Abbasi, and F. Mokhtarian, "Affine-Similar Shape Retrieval: Application to Multi-View 3-D Object Recognition", IEEE Transactions on Image Processing, Volume 10, Number 1, pp. 131-139, 2001.
- [3] C. Dorai and A.K. Jain, "Shape Spectrum Based View Grouping and Matching of 3D Free-Form Objects", IEEE Transactions on PAMI, Volume 19, Number 10, pp.1139-1146, 1997.
- [4] S. Mahmoudi et M. Daoudi, "Une nouvelle méthode d'indexation 3D", 13ème Congrès Francophone de Reconnaissance des Formes et Intelligence Artificielle (RFIA2002), volume 1, pp. 19-27, Angers, France, 8-9 janvier 2002.
- [5] J-L. Chen and G. Stockman, "3D Free-Form Object Recognition Using Indexing by Contour Feature", Computer Vision and Image Understanding, Volume 71, Number 3, pp.334-355, 1998.
- [6] J.H. Yi and D.M. Chelberg, "Model-Based 3D Object Recognition Using Bayesian Indexing", in Computer Vision and Image Understanding Volume 69, Number 1, January, pp.87-105,1998.
- [7] J.J. Koenderink and A.J. van Doorn, "Surface shape and curvature scales", Image and Vision Computing, Volume 10, Number 8, pp. 557-565, 1992.
- [8] C. Nastar, "The image Shape Spectrum for Image Retrieval", INRIA Technical Report number 3206, 1997.
- [9] T. Zaharia et F. Prêteux, "Indexation de maillages 3D par descripteurs de forme", 13ème Congrès Francophone de Reconnaissance des Formes et Intelligence Artificielle (RFIA2002), volume 1, pp. 48-57, Angers, France, 8-9 janvier 2002.